

Exact common-activity compatibility for autocatalytic junction–path assemblies: boundary reduction, a rational decision algorithm, and operating certificates

14 September 2026

Abstract

Autocatalytic cores that are each realizable in isolation need not be realizable together: under thermodynamically consistent mass action, every reaction that involves a species sees the same activity of that species, and the stoichiometric powers of the activities constrain both the direction and the magnitude of every current. Kosci, Kuperberg, Rajon and Charlat (*Proc. Natl. Acad. Sci. USA*, 2025) proved that every isolated potential autocatalytic core admits a productive realization and exhibited two cores with no common one. We study a class of assemblies in which the number of species is unbounded and compatibility is nevertheless decided exactly. An assembly consists of weighted two-reaction cores $X_u \rightleftharpoons X_v$, $X_v + F \rightleftharpoons 2X_u$ with arbitrary positive kinetic factors, arranged in directed paths whose interiors are private and whose endpoints belong to a set of k shared junction species, each species carrying a closed activity box. We prove that a common activity vector making every selected core strictly productive exists if and only if the junction activities alone satisfy, for each path, a strict inequality between two explicitly composed quadratic responses, together with the junction boxes. The interior of every path is eliminated without relaxing a single current law, and a one-parameter interpolation reconstructs it. For rational data this yields a complete decision procedure that also outputs a rational common state; its total bit cost and output length are at most $\mathcal{A}^{O(k+1)}$ with $\mathcal{A} = (n + M + k + 2)(2^h + 1)(B + h + 1)$, where n is the input size, M the number of paths, h their maximum length and B the rational height of the data. The precision analysis, which combines a reciprocal root bound for a one-block quantifier-elimination output with Lipschitz control of the composed responses, replaces algebraic sample points by dyadic bisection and handles singleton junction boxes. We further give a linear-programming sufficient certificate that produces explicit compatible assemblies without any integer grading, a sharp fixed-state tolerance radius for independent relative factor uncertainty, an eight-corner certificate for joint activity, factor and food uncertainty, exact food-activity windows, and instantaneous loss budgets. The source-level equivalence, the path reconstruction, the merge across shared species, the rational certificate checker and every local certificate lemma are verified in Lean 4 against Mathlib with warnings promoted to errors and no admitted statements; the quantifier-elimination instantiation, the bit-size accounting and the linear-programming complexity are conventional proofs and are marked as such.

1 Introduction

A *potential autocatalytic core* (PAC) is a minimal motif of a reaction network that admits a strictly productive reaction current [1, 2]. Whether such a current can be realized by an actual concentration vector is a separate question. Under mass action with fixed transition-state factors the current of a reversible reaction is a positive multiple of the difference of its reactant and product complex activities, and a complex activity is a monomial in the species activities.

Kosc, Kuperberg, Rajon and Charlat proved that every isolated PAC is realizable in this sense and showed that two cores sharing a species may admit a common productive current but no common concentration vector [2]. Two predecessor papers quantified this phenomenon: an exact phase diagram for a pair of overlapping triangles and a layered account of the failure mechanisms [3], and, for families of two-reaction cores, minimally incompatible families of every order, an exact univariate elimination for bounded fans, and a polynomial-time constructive certificate for *graded* assemblies, in which integer ranks increase by one along every core [4]. The graded certificate is sufficient only, the fan theorem fixes a single shared reaction, and the unbounded-order theorem shows that no test based on bounded subfamilies can decide compatibility in general.

This paper isolates a class of assemblies that sits between those results: the number of species and of cores is unbounded, the kinetic factors are arbitrary positive reals that differ from core to core, integer grading is not assumed, and yet compatibility is decided exactly with a rational witness. The class consists of *junction-path assemblies*: the selected cores are partitioned into directed paths whose interior species are private to one path and whose endpoints belong to a designated set of k junction species (Figure 1). Junctions may carry any number of paths and form an arbitrary skeleton. Each species carries a closed activity box; junction boxes may be singletons.

Main results. Write each selected core $e: u \rightarrow v$ as $X_u \rightleftharpoons X_v$, $X_v + F \rightleftharpoons 2X_u$ with positive factors a_e, b_e and food activity one, so that its two currents are $p_e = a_e(x_u - x_v)$ and $q_e = b_e(x_v - x_u^2)$, and call it productive when both of its internal species are net produced, $2q_e - p_e > 0$ and $p_e - q_e > 0$.

- (A) *Boundary reduction* (Theorems 4.1 and 5.1). A single core is productive exactly when $G_e(x_u) < x_v < F_e(x_u)$ for two explicit quadratic responses. Along a path the responses compose in physical order, and every endpoint value strictly between the composed responses is attained by a productive interior whose values stay between the endpoints. Consequently a common productive state in the prescribed boxes exists if and only if the k junction activities satisfy the composed strict inequalities and their boxes. No current law is relaxed, and no species receives two activities.
- (B) *Exact rational algorithm* (Theorem 6.1). For rational factors and boxes the reduced system is decided by one block of k existentially quantified variables, and a rational common state is reconstructed by dyadic bisection with a proved precision bound. The total bit cost and output length are at most $\mathcal{A}^{O(k+1)}$ with $\mathcal{A} = (n + M + k + 2)(2^h + 1)(B + h + 1)$. For fixed k and $h = O(\log n)$ this is polynomial in the binary input size.
- (C) *A linear sufficient class* (Proposition 7.1). Near the equilibrium point $x \equiv 1$ the exact residual identities reduce to a homogeneous strict linear system in *deficits*, decided by a rational linear program. This produces compatible assemblies that admit no integer grading, for instance a nine-species unit-factor assembly of two private paths of lengths four and five.
- (D) *Operating certificates* (Section 8). At a fixed productive state we give the sharp radius ρ_* of independent relative factor uncertainty, an eight-corner certificate for joint activity, factor and food rectangles, the exact food-activity window, and instantaneous loss budgets, all as closed rational formulas.

Theorem 5.1 is a statement about *existence* for arbitrary positive real factors; rationality and the parameters k, h enter only in the algorithm. The obstruction examples of [4] sit inside the class: a directed cycle is never compatible, and the unit-factor path with a shortcut is never compatible, whereas the same shortcut graph with factors $(9, 1), (9, 1), (2, 1)$ has an

explicit rational common state (Section 8). Kinetic factors therefore change the answer without changing the PAC status or the architecture of the cores.

Method. The one-edge inequality is a rearrangement of the literal currents. Its composition along a path uses only that both responses are strictly increasing on $[0, 1]$ and fix 0 and 1. Sufficiency is the substantive step: we interpolate between the lower and upper response of every edge with one common parameter λ and apply the intermediate value theorem to the endpoint. This constructs a productive interior with the prescribed final activity exactly, which is what allows the interior to be discarded and later rebuilt with rational values. The algorithmic part rests on a published one-block quantifier-elimination bound [5, Theorem 2.27], [6], used in two roles: as the decision routine, and, with one free slack variable, as a way to bound the size of a positive residual that some feasible point must have. A reciprocal Cauchy bound on the coefficients of the eliminated univariate formula gives a dyadic slack in $H + 3$ trials; the composed responses are 2^h -Lipschitz, which turns that slack into a bracket width; bisection of closed brackets keeps a common real witness and finally yields rational midpoints whose residuals stay positive. Private values are recovered by bisection in the interpolation parameter with an explicit last-edge margin, without root isolation or a number-field compositum.

What is verified. The chemical equivalence of Theorem 5.1 is a Lean 4 theorem stated on the literal reversible network with activity monomials and global species identities, and the rational state checker, the examples and all local certificate lemmas of Sections 7 and 8 are compiled in the same development [7, 8]. Theorem 6.1 and the linear-programming complexity are conventional proofs building on cited algorithms. Section 9 lists the boundary precisely.

What is not claimed. We decide compatibility for a supplied decomposition; we do not find an optimal one. Private species share a common box $[\ell, 1]$; arbitrary independent private boxes are outside the theorem, and Remark 4.2 shows why. The results are static: they concern the existence of one activity vector at which every selected core is productive, not persistence, steady states or long-time behaviour of a dynamical system. The complexity bound is not a fixed-parameter-tractability result in k and the exponential degree 2^h of the expanded responses is a property of one representation, not a lower bound.

Organization. Section 2 fixes the network, the cores and the class. Section 3 proves the one-edge interval and its regularity, Section 4 the path reconstruction, and Section 5 the assembly theorem. Section 6 gives the rational algorithm and its bit bound, with a small executable demonstration. Section 7 treats the linear sufficient class and the nongraded example, Section 8 the operating certificates at the weighted example, Section 9 the formal verification, and Section 10 the limits and open questions.

2 Networks, cores and the junction–path class

2.1 Reversible networks with fixed barrier factors

A reversible chemical reaction network on finite species and reaction index sets assigns to every reaction r a reactant complex and a product complex, each a vector of nonnegative integers over the species, and a positive barrier factor β_r . For an activity vector z the complex activity of c is the monomial $\prod_s z_s^{c_s}$, and the *current* of r is

$$v_r(z) = \beta_r \left(\prod_s z_s^{c_{r,s}^{\text{react}}} - \prod_s z_s^{c_{r,s}^{\text{prod}}} \right). \quad (1)$$

This is mass action at fixed temperature with the standard chemical potentials chosen so that every displayed forward/reverse equilibrium ratio equals one; the factor β_r multiplies both directions and does not alter the equilibrium ratio. A *motif* is a pair (set of species, set of reactions); it is *productive* for a current vector v if every species of the motif has strictly positive net production $\sum_r S_{s,r} v_r$, where S is the stoichiometric matrix, and it is a PAC if it is side-incident, admits some productive current, and no strict submotif does [2]. These are the definitions formalized in the predecessor development [3]; PAC status depends only on the reactant and product complexes, not on the barrier factors, and this is a compiled lemma (Section 9).

2.2 Weighted two-reaction cores

Let a finite oriented simple graph have species set V and selected edge set E : no loops, no duplicate edges, no antiparallel pairs. An edge $e: u \rightarrow v$ owns two reversible channels,

$$X_u \rightleftharpoons X_v, \quad X_v + F \rightleftharpoons 2X_u, \quad (2)$$

with factors $a_e, b_e > 0$. With the food species F held at activity one, (1) gives the two oriented currents

$$p_e = a_e(x_u - x_v), \quad q_e = b_e(x_v - x_u^2). \quad (3)$$

The net production of the two internal species of the core is

$$r_{e,u} = 2q_e - p_e, \quad r_{e,v} = p_e - q_e. \quad (4)$$

The core is *productive* when both residuals are strictly positive; equivalently $p_e/2 < q_e < p_e$, which forces $p_e, q_e > 0$. Each core is a PAC [4], and the family of all selected cores is *compatible* in a box if one activity vector in the box makes every selected core productive. Ownership of channels is part of the input: two edges never share a physical reaction, and no core is required to be productive merely as part of a sum with other cores.

Physical normalization. Activities x_s and f are dimensionless relative to stated reference concentrations; ideality identifies them with normalized concentrations. A common reference concentration c_* and flux scale J_* make time dimensionless in units c_*/J_* ; the coefficients and residuals below use this normalized flux unit. For a mass-balanced schematic realization, F and all X_s must have the same elemental composition and charge, for instance distinct isomers of a common unit; equal molecular mass alone would not balance the doubling channel. No specific metabolite, enzyme mechanism or calibrated parameter range is assumed; the factors are model conductances, not measured constants. The constraints of (1) are the loop-law type constraints of [9] specialized to a fixed equilibrium ratio.

2.3 Junction–path assemblies

Definition 2.1 (Admitted decomposition). Fix $0 < \ell < 1$. A *junction–path assembly* is the graph above together with a designated set $J \subseteq V$ of k *junctions* and a partition of E into M nonempty directed paths $P_1, \dots, P_M$ of lengths at most h such that

- (i) both endpoints $s(P), t(P)$ of every path are junctions;
- (ii) every interior vertex of a path is *private*: it is not a junction and occurs on exactly one path, exactly once.

Each junction j carries a closed box $[l_j, u_j] \subseteq [\ell, 1]$, possibly a singleton; every private species and every isolated species carries the common box $[\ell, 1]$.

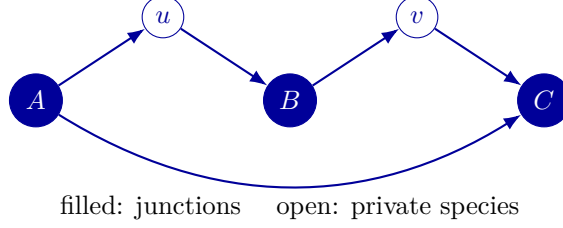


Figure 1: A junction–path assembly with $k = 3$ junctions and $M = 3$ paths. Each arrow is a two-channel core (2), not a single elementary reaction. The two upper paths have private interiors; the lower path is a direct junction-to-junction core. A filled vertex has one activity in every incident path.

Species that lie on no selected edge are permitted and play no role. The decomposition and the boxes are explicit data to be validated; we do not assume that a decomposition with few junctions is supplied or can be found cheaply. A junction may be the endpoint of arbitrarily many paths, and paths of length one are direct junction-to-junction cores.

Relation to the predecessors. The graded assemblies of [4] are pair assemblies with unit factors admitting integer ranks that increase by one along every edge; the theorem there is sufficient only. The fans of [4] and the triangle pairs of [3] share one reaction and reduce to one variable. The present class has arbitrarily many species and paths, arbitrary factors, and k nonlinearly interacting junction variables; unequal path lengths between the same junctions are permitted, which rules out grading (Sections 7 and 8). The parameter k is the number of remaining variables after the paths are eliminated. It is not the width of a tree decomposition of the graph, and the class is not a bounded-treewidth class.

3 The one-edge response interval

For an edge with factors $a, b > 0$ define the *lower* and *upper responses*

$$G(x) = \frac{ax + 2bx^2}{a + 2b}, \quad F(x) = \frac{ax + bx^2}{a + b}. \quad (5)$$

Lemma 3.1 (One-edge interval). *For all real x, y the core is productive at $(x_u, x_v) = (x, y)$ if and only if $G(x) < y < F(x)$. Moreover:*

- (a) *G and F are convex combinations of x and x^2 ; they map $[0, 1]$ into $[0, 1]$, fix 0 and 1, and are strictly increasing on $[0, \infty)$;*
- (b) *for $0 \leq x \leq y \leq 1$ and $c \in \{1, 2\}$, with $R_c(x) = (ax + cbx^2)/(a + cb)$,*

$$R_c(y) - R_c(x) = (y - x) \frac{a + cb(x + y)}{a + cb}, \quad 0 < \frac{a + cb(x + y)}{a + cb} < 2, \quad (6)$$

so both responses are 2-Lipschitz on $[0, 1]$;

- (c) *for $0 < x < 1$,*

$$x^2 < G(x) < F(x) < x, \quad F(x) - G(x) = \frac{abx(1 - x)}{(a + b)(a + 2b)} > 0. \quad (7)$$

Productivity implies $x_v < x_u$, and the answer depends on a, b only through the ratio a/b .

Proof. Direct rearrangement of (4) gives the two identities

$$2b(y - x^2) - a(x - y) = (a + 2b)(y - G(x)), \quad a(x - y) - b(y - x^2) = (a + b)(F(x) - y),$$

with positive prefactors, which is the equivalence. Part (a) is immediate from the form of R_c , and (b) is a direct computation whose quotient lies in $(0, (a + 2cb)/(a + cb)) \subseteq (0, 2)$. For (c), $R_c(x) - x^2 = ax(1 - x)/(a + cb) > 0$ and $x - R_c(x) = cbx(1 - x)/(a + cb) > 0$, and the displayed difference is a direct computation. Positivity of $p_e = a(x - y)$ follows from $y < F(x) < x$. Multiplying a and b by a common positive constant leaves G and F unchanged. $\square$

Scaling both factors of a core rescales its currents but not its productivity; static compatibility of a family therefore depends only on the factor ratios a_e/b_e , while the residual magnitudes in Section 8 do not.

4 Exact elimination along a private path

For a path $P = (e_1, \dots, e_r)$ in physical order put

$$G_P = G_{e_r} \circ \dots \circ G_{e_1}, \quad F_P = F_{e_r} \circ \dots \circ F_{e_1}. \quad (8)$$

Both are strictly increasing on $[0, \infty)$, fix 0 and 1, and are 2^r -Lipschitz on $[0, 1]$ by Lemma 3.1. A *realization* of P with initial value x and final value y is a sequence $x = t_0, t_1, \dots, t_r = y$ such that edge e_i is productive at (t_{i-1}, t_i) ; it is *boxed* in $[\ell, 1]$ when all $t_i \in [\ell, 1]$.

Theorem 4.1 (Fixed-endpoint path reconstruction). *Let $\ell \leq x \leq 1$ and $\ell \leq y$. The path P has a boxed realization with initial value x and final value y if and only if*

$$G_P(x) < y < F_P(x). \quad (9)$$

When (9) holds, a realization exists with $x = t_0 > t_1 > \dots > t_r = y$, so that every interior value lies strictly between the endpoints.

Proof. Necessity. A realization has $t_i > 0$ for all i , since $q_{e_i} > 0$ forces $t_i > t_{i-1}^2 \geq 0$. Lemma 3.1 gives $G_{e_1}(x) < t_1 < F_{e_1}(x)$, and inductively, using strict monotonicity of $G_{e_{i+1}}$ and $F_{e_{i+1}}$ on $[0, \infty)$,

$$G_{e_{i+1}}(G_{e_i} \circ \dots \circ G_{e_1}(x)) < G_{e_{i+1}}(t_i) < t_{i+1} < F_{e_{i+1}}(t_i) < F_{e_{i+1}}(F_{e_i} \circ \dots \circ F_{e_1}(x)).$$

Sufficiency. Since $G_P(1) = F_P(1) = 1$, the strict band (9) excludes $x = 1$, so $0 < \ell \leq x < 1$. For $\lambda \in [0, 1]$ define the blended response $H_i(\lambda, t) = (1 - \lambda)G_{e_i}(t) + \lambda F_{e_i}(t)$ and the orbit

$$t_0(\lambda) = x, \quad t_i(\lambda) = H_i(\lambda, t_{i-1}(\lambda)) \quad (1 \leq i \leq r).$$

The endpoint $\lambda \mapsto t_r(\lambda)$ is continuous with $t_r(0) = G_P(x)$ and $t_r(1) = F_P(x)$, so by the intermediate value theorem there is $\lambda \in (0, 1)$ with $t_r(\lambda) = y$. For $0 < t < 1$ and $0 < \lambda < 1$ the blend $H_i(\lambda, t)$ lies strictly inside $(G_{e_i}(t), F_{e_i}(t))$ because the gap (7) is positive, and $0 < H_i(\lambda, t) < t < 1$ by (7). Inductively every $t_i(\lambda) \in (0, 1)$, every edge is productive, and $x = t_0(\lambda) > t_1(\lambda) > \dots > t_r(\lambda) = y$. All interior values lie in $(y, x) \subseteq [\ell, 1]$. $\square$

Theorem 4.1 includes one-edge paths, singleton endpoint constraints and boundary equality, which is infeasible. The interpolation orbit is also the object bisected in Section 6 to obtain rational interior values.

Remark 4.2 (Why the private box must be common). The theorem assigns interior values between the endpoints, and any realization must do so, since productivity decreases activity along every edge. An individual box on an interior species that lies entirely above the initial value x , or below the final value y , therefore makes the path infeasible even when (9) holds. Arbitrary private boxes thus cannot be handled by the same two-function reduction; they require additional junction-like variables. This was confirmed on an exact solver control before the theorem was formalized.

Proposition 4.3 (Two obstructions).

- (i) *A directed cycle of selected cores has no common productive state in the positive orthant, for every choice of positive factors.*
- (ii) *With unit factors, the path $A \rightarrow B \rightarrow C$ and the shortcut $A \rightarrow C$ have no common productive state in the positive orthant. Writing $G(x) = (x+2x^2)/3$ and $F(x) = (x+x^2)/2$,*

$$G(x) - F(F(x)) = \frac{x(1-x)(3x^2+9x+2)}{24} > 0 \quad (0 < x < 1). \quad (10)$$

Proof. (i) Productivity decreases activity along every edge, and a cycle would return to its starting species. (ii) A productive edge with positive activities forces $x_A < 1$ by (7). The two-edge path requires $x_C < F(F(x_A))$ by Theorem 4.1, the shortcut requires $x_C > G(x_A)$, and (10) is an exact polynomial identity. $\square$

Part (ii) is the $n = 3$ case of the unbounded-order conflict of [4]; it is an exact negative control, not a failure of a local numerical search. Acyclicity alone is therefore not sufficient for compatibility, and Section 8 shows that the same graph becomes compatible for other factors.

5 The junction–path compatibility theorem

Theorem 5.1 (Junction–path compatibility). *Let a junction–path assembly with arbitrary positive real factors and boxes $[l_j, u_j] \subseteq [\ell, 1]$ be given. There is an activity vector $x \in \mathbb{R}^V$ with $x_s \in [\ell, 1]$ for all s , $x_j \in [l_j, u_j]$ for all $j \in J$, and every selected core productive, if and only if there is a vector $z \in \mathbb{R}^J$ with*

$$l_j \leq z_j \leq u_j \quad (j \in J), \quad G_P(z_{s(P)}) < z_{t(P)} < F_P(z_{s(P)}) \quad (P = P_1, \dots, P_M). \quad (11)$$

The inequalities (11) are inequalities on the actual shared species: a junction has one variable, not one copy per incident path.

Proof. If x is a common productive state, its restriction to each path is a boxed realization between the junction values, so Theorem 4.1 gives (11) with $z = x|_J$. Conversely fix a feasible z . For each path apply Theorem 4.1 with initial value $z_{s(P)}$ and final value $z_{t(P)}$; both lie in $[\ell, 1]$ because the junction boxes are subboxes of $[\ell, 1]$. Define $x_j = z_j$ on junctions and x_s equal to the constructed value of the unique occurrence of s on the unique path containing a private species s ; give isolated species the value ℓ . Condition (ii) of Definition 2.1 makes this single-valued, and every occurrence of a junction on a path is an endpoint whose constructed value equals z_j by construction. Every selected core is productive at x and all boxes hold. Finally, the identities (3)–(4) identify the edge statements with the source-level predicate that the motif of e is productive for the current vector (1) of the literal network, and every motif is a PAC. $\square$

The identification in the last sentence, from the abstract edge inequalities back to the reversible network with activity monomials, is where the formal statement lives: the Lean

theorem `sourceCompatible_iff_junctionFeasible` (Section 9) is stated for a structure that records the paths, the placement of path positions on species, the junction predicate, the endpoint characterization and private uniqueness, and asserts the equivalence between the source-level compatibility predicate and the feasibility of (11). The merge step is a separate compiled lemma that produces one global vector from local path vectors agreeing on junctions.

Corollary 5.2. *For $k = 0$ a valid assembly has no paths, and every vector in the boxes is compatible. For $k \geq 1$ compatibility is a semialgebraic condition on $z \in \mathbb{R}^k$ defined by $2M$ strict polynomial inequalities of degree at most 2^h and $2k$ weak linear ones.*

6 A complete rational algorithm with a bit bound

Assume now that all factors, ℓ and all box endpoints are rational, let $B \geq 1$ bound the bit lengths of their numerators and denominators, and let n bound the explicit input size (species, edges, path lists and their labels). Put $D = 2^h$.

6.1 The external decision routine

We use the one-block quantifier-elimination theorem of Basu, Pollack and Roy in the form stated in Basu’s survey [5, Theorem 2.27], [6, Chapter 14]: for a formula with at most s polynomials of degree at most d in one block of k quantified variables and ℓ_f free variables, an equivalent quantifier-free formula is computed with $s^{(k+1)(\ell_f+1)} d^{O(k)O(\ell_f)}$ arithmetic operations, and if the coefficients are integers of bit size at most τ , then all intermediate and output integers have bit size at most $\tau d^{O(k)O(\ell_f)}$. We use $\ell_f \in \{0, 1\}$; a pure decision may be treated as having one unused free variable, so that in both roles the operation count is $(s(d+1))^{O(k+1)}$ and the integer height is $\tau(d+1)^{O(k+1)}$. Earlier singly exponential procedures [10, 11] would serve equally; cylindrical algebraic decomposition [12] and Tarski’s original method [13] are complete but have worse bounds. This is the only external algorithm the theorem depends on, and the contribution here is the reduction to it and the control of the rational output, not the elimination itself.

6.2 The algorithm

Let $f_1, \dots, f_{2M}$ denote the $2M$ response residuals $z_{t(P)} - G_P(z_{s(P)})$ and $F_P(z_{s(P)}) - z_{t(P)}$, with positive denominators cleared so that each is an integer polynomial in z .

Step 1. *Validate.* Check the graph, factors, boxes, path orientation and continuity, endpoint membership in J , private uniqueness and edge coverage; reject malformed data. If $E = \emptyset$, return box midpoints for junctions and ℓ elsewhere.

Step 2. *Reduce.* Construct G_P, F_P for every path by rational polynomial composition (Lemma 6.2). Decide (11). If infeasible, return INCOMPATIBLE.

Step 3. *Slack.* For $t = 1, \frac{1}{2}, \frac{1}{4}, \dots$ decide whether the boxes together with $f_i \geq t$ for all i are feasible; keep the first success t_0 (Lemma 6.3).

Step 4. *Junction brackets.* Maintain one closed rational bracket per junction, initially its box, with the invariant that some real vector in the product of brackets satisfies all $f_i \geq t_0$. Repeatedly bisect a bracket and decide whether its left closed half, the other current brackets and all $f_i \geq t_0$ are jointly feasible; keep the left half if so, otherwise the right half. Stop when all widths are at most $\eta = t_0/(4(D+1))$ (Lemma 6.4).

Step 5. *Private reconstruction.* Take the bracket midpoints as junction activities. For each path of length at least two, bisect the interpolation parameter of Theorem 4.1 with

exact rational arithmetic until the last edge is productive with the prescribed final value (Lemma 6.5).

Step 6. *Check and output.* Merge by species identity, evaluate every current and residual exactly, verify every box, and output the rational state.

Theorem 6.1 (Complete rational decision and reconstruction). *Steps 1–6 decide every validated rational instance and, when feasible, output a rational common productive state. The total bit cost and the output length are bounded by*

$$\mathcal{A}^{O(k+1)}, \quad \mathcal{A} = (n + M + k + 2)(2^h + 1)(B + h + 1). \quad (12)$$

For fixed k and $h \leq C \log_2(n + 2)$ with fixed C this is polynomial in the binary input size.

The remainder of this section proves the four lemmas that the steps rely on and then assembles the bound. Table 1 collects the quantities.

6.3 Response representation

Lemma 6.2 (Positive representation). *Write $a = A_1/A_2$ and $b = B_1/B_2$ in positive integers and put $\alpha = A_1B_2$, $\beta = B_1A_2$. Represent a composed response as $N(X)/Q$ with an integer polynomial N and a positive integer Q . Appending an edge with lower ($c = 2$) or upper ($c = 1$) response gives*

$$N_{\text{new}} = \alpha Q N + c\beta N^2, \quad Q_{\text{new}} = (\alpha + c\beta)Q^2. \quad (13)$$

Starting from $N = X$, $Q = 1$, every numerator has nonnegative coefficients summing to Q ; hence each coefficient is at most Q . After r edges the degree is exactly 2^r , and if all input numerators and denominators are below 2^B then $\log_2 Q < (2^r - 1)(2B + 2)$. Consequently every composed response has degree at most D and integer height $\tau = O(D(B + 1))$, and constructing both responses of every path by dense schoolbook arithmetic costs a polynomial in M , D and B . Clearing the positive denominators turns (11) into $s = O(M + k + 1)$ integer polynomial constraints in k variables of degree at most D and height $O(\tau)$, in a formula of length linear in s .

Proof. Substituting N/Q into $(\alpha x + c\beta x^2)/(\alpha + c\beta)$ gives (13). If N has nonnegative coefficients with $N(1) = Q$, then N_{new} has nonnegative coefficients and $N_{\text{new}}(1) = \alpha Q^2 + c\beta Q^2 = Q_{\text{new}}$. Leading coefficients are positive, so the degree doubles. From $\alpha + c\beta < 2^{2B+2}$ the recurrence $\log_2 Q_{\text{new}} = \log_2(\alpha + c\beta) + 2 \log_2 Q$ gives the stated bound. The remaining statements are direct. $\square$

The representation is not reduced by greatest common divisors, and the bound is a property of this representation: for $a = b = 1$ and the upper response the reduced leading coefficient after r edges is $2^{-(2^r-1)}$, so the exponential height is intrinsic to the expanded polynomial, while no hardness for other algorithms follows.

6.4 A dyadic slack with a proved search bound

Lemma 6.3 (Slack). *Suppose (11) is feasible. Then there is a universal constant c_0 such that with*

$$H = \tau(D + 1)^{c_0(k+1)} \quad (14)$$

every $t \in (0, 2^{-(H+2)})$ is a feasible slack, i.e. the boxes and $f_i \geq t$ for all i are jointly satisfiable. Hence Step 3 succeeds within $H + 3$ decisions.

Proof. Consider the formula with quantified block $z \in \mathbb{R}^k$ and one free variable t : the boxes, $f_i(z) \geq t$ for all i , and $0 < t \leq 1$. By the backend theorem an equivalent quantifier-free formula $\Psi(t)$ exists whose polynomials have integer coefficients of bit size at most H as in (14), for a suitable universal c_0 . Discard identically zero polynomials, whose signs are constant, and divide every remaining polynomial by its largest power of t , which does not change signs on $t > 0$; its constant coefficient is then a nonzero integer of absolute value at least one. By the reciprocal Cauchy bound [14], every nonzero root of an integer polynomial with constant term $c'_0 \neq 0$ and coefficients of absolute value at most 2^H has absolute value at least $|c'_0|/(|c'_0| + 2^H) > 2^{-(H+1)}$. Hence no polynomial of Ψ changes sign on $(0, 2^{-(H+2)})$ and the truth value of Ψ is constant there. The set of feasible slacks is nonempty, since a strictly feasible z has a positive minimum residual, and it is downward closed in $(0, 1]$; so it meets $(0, 2^{-(H+2)})$, and by constancy contains it. The dyadic sequence 2^{-i} enters that interval at $i = H + 3$. $\square$

The elimination with a free variable is used only to bound the number of decisions actually executed. The algorithm itself never constructs Ψ , isolates a root or guesses a chemical margin.

6.5 Rational junction coordinates

Lemma 6.4 (Junction precision). *Each composed response is D -Lipschitz on $[0, 1]$, so every residual f_i has Lipschitz constant at most $D + 1$ with respect to the maximum norm on the box. Step 4 preserves the invariant that one real vector in the product of the current brackets has all residuals at least t_0 ; when all widths are at most $\eta = t_0/(4(D + 1))$, the vector z° of bracket midpoints satisfies every residual $f_i(z^\circ) \geq 3t_0/4 > 0$ and lies exactly in every original box, including singleton boxes. Each coordinate needs $O(H + h + 1)$ decisions, and the rational midpoints have bit length $L = O(B + H + h + 1)$.*

Proof. The Lipschitz bound composes (6) at most h times. If the left closed half of a bracket, together with the other brackets and the residual constraints, is infeasible, then the witness guaranteed by the invariant lies in the right closed half, because the two closed halves cover the parent bracket; so the invariant is preserved either way, and it holds initially by Step 3. At the end some real witness w lies in all final brackets, so $\|z^\circ - w\|_\infty \leq \eta$ and $f_i(z^\circ) \geq t_0 - (D + 1)\eta = 3t_0/4$. Midpoints of closed subbrackets of the boxes lie in the boxes; a singleton box is never bisected. A bracket of initial width at most one reaches width η after $\lceil \log_2(1/\eta) \rceil = O(\log(1/t_0) + h) = O(H + h + 1)$ bisections by Lemma 6.3, and its endpoints are dyadic refinements of B -bit rationals. $\square$

Retaining brackets rather than fixing a rational value for one coordinate before the others are refined is essential: a prematurely fixed coordinate may destroy the common witness for the remaining strict inequalities.

6.6 Rational private reconstruction

Lemma 6.5 (Rational bisection). *Let P have length $r \geq 2$ and rational endpoints $\ell \leq y < x < 1$ with $G_P(x) < y < F_P(x)$. Put*

$$\delta = \min\{y - G_P(x), F_P(x) - y\} > 0, \quad C = 2^r - 1, \quad C' = 2^{r-1} - 1.$$

Let $E(\lambda) = t_r(\lambda)$ and $T(\lambda) = t_{r-1}(\lambda)$ be the full and prefix endpoints of the interpolation orbit of Theorem 4.1, and let a, b be the factors of the last edge. Then

- (a) *E and T are strictly increasing on $[0, 1]$ with Lipschitz constants C and C' in λ ;*
- (b) *if $E(\lambda_*) = y$ then $\lambda_*, 1 - \lambda_* \geq \delta/C$, $y < T(\lambda_*) < x$, and the last-edge gap at $T(\lambda_*)$ is at least $\gamma = ab\ell(1 - x)/((a + b)(a + 2b)) > 0$, so both last-edge margins $y - G(T(\lambda_*))$ and $F(T(\lambda_*)) - y$ are at least $m = \delta\gamma/C$;*

(c) every rational $\lambda \in (0, 1)$ with $|\lambda - \lambda_*| < m/(4C')$ makes all prefix edges productive along the orbit and the last edge productive at $(T(\lambda), y)$.

Consequently bisection of $[0, 1]$ on the sign of $E(\lambda) - y$, testing the last edge with the prescribed y at each midpoint, terminates within $1 + \lceil \log_2(4CC'/(\delta\gamma)) \rceil = O(D(B+L) + h + 1)$ iterations, and the private values it produces have bit length $O(D^2(B+L+h+1))$.

Proof. (a) Each blend $H_i(\lambda, t)$ is 2-Lipschitz in t by (6) and 1-Lipschitz in λ because $0 \leq F_{e_i}(t) - G_{e_i}(t) \leq 1$ on $[0, 1]$; induction gives the constants $\sum_{i < r} 2^i = C$ and C' . The first blend is strictly increasing in λ for $x \in (0, 1)$ by (7), and every later map is strictly increasing in its input and nondecreasing in λ . (b) From $E(0) = G_P(x)$, $E(1) = F_P(x)$ and the Lipschitz bound, $\delta \leq |E(\lambda_*) - E(0)| \leq C\lambda_*$, and likewise for $1 - \lambda_*$. The orbit is strictly decreasing, so $y < T(\lambda_*) < x$, and (7) evaluated at $T(\lambda_*) \in [\ell, x]$ is at least γ . Since $y = (1 - \lambda_*)G(T(\lambda_*)) + \lambda_*F(T(\lambda_*))$, the two margins are λ_* and $1 - \lambda_*$ times the gap. (c) The functions $\lambda \mapsto G(T(\lambda))$ and $F(T(\lambda))$ are $2C'$ -Lipschitz, so within $m/(4C')$ of λ_* both last-edge inequalities lose less than $m/2$ of their margin and stay strict; prefix edges are productive for every $\lambda \in (0, 1)$ by Theorem 4.1. Bisection retains a bracket containing λ_* ; once its length is below $m/(4C')$ its midpoint passes, and equality $E(\lambda) = y$ passes immediately. The rational numbers defining δ have bit length $O(D(B+L))$ by Lemma 6.2, γ has bit length $O(B+L)$, and $\log_2(CC') = O(h)$, which gives the iteration bound. A bisection parameter has that many bits, and evaluating the quadratic orbit $r \leq h$ times multiplies bit lengths by at most $2^r \leq D$. $\square$

One-edge paths need no reconstruction. The lemma replaces root isolation in a compositum of the path fields by exact bisection with an explicit last-edge margin.

6.7 Assembling the bound

Proof of Theorem 6.1. Correctness. Step 2 is Theorem 5.1 together with Corollary 5.2. If it succeeds, Step 3 terminates by Lemma 6.3, Step 4 produces junction values in the boxes with all residuals positive by Lemma 6.4, and Step 5 produces a boxed realization of every path with those endpoints by Lemma 6.5. The merge of Theorem 5.1 is single-valued, and Step 6 is an exact recheck of the literal currents, boxes and residuals, so the output is a certified common productive state. If Step 2 fails, no common state exists by Theorem 5.1.

Cost. The number of decisions in Steps 2–4 is $Q_* = O((k+1)(H+B+h+1))$ by Lemmas 6.3 and 6.4. Every decision has $O(M+k+1)$ polynomials of degree at most D with integer coefficients of bit size

$$T_q = O(\tau + B + H + h + 1),$$

which includes the cleared response denominators, the current slack and the current bracket endpoints. By the backend bound a decision costs at most $(s(D+1))^{c(k+1)} [T_q(D+1)^{c'(k+1)}]^2$ bit operations for universal c, c' , the square accounting for schoolbook integer arithmetic on intermediate integers of that size. The height T_q enters with a fixed exponent; substituting (14) and multiplying by Q_* therefore leaves an exponent $O(k+1)$ rather than $O(k^2)$. Validation is polynomial in n and B with a sorting step for edge coverage; response construction is polynomial in M, D, B by Lemma 6.2; private reconstruction of M paths is polynomial in D, B, L by Lemma 6.5; the exact check and the output of $O(n)$ rational values of bit length $O(D^2(B+L+h+1))$ are polynomial in the same quantities. All of these are bounded by (12). For $k = 0$ a valid assembly has no paths, and the empty-edge case is handled in Step 1 without any decision. For fixed k and $h \leq C \log_2(n+2)$, D is polynomial in n and the exponent is fixed. $\square$

Quantity	Bound
Response degree; integer height	$D = 2^h; \tau = O(D(B+1))$
Constraints per decision	$s = O(M+k+1)$
Slack-elimination height	$H = \tau(D+1)^{O(k+1)}$
Dyadic slack	$t_0 > 2^{-(H+3)}$
Decisions in Steps 2–4	$Q_* = O((k+1)(H+B+h+1))$
Integer height per decision	$T_q = O(\tau+B+H+h+1)$
Junction bits per coordinate	$L = O(B+H+h+1)$
Bisection iterations per path	$O(D(B+L)+h+1)$
Private bits per coordinate	$O(D^2(B+L+h+1))$

Table 1: Precision and size accounting for Theorem 6.1. Heights count bits of numerators and denominators or of integer coefficients, as appropriate.

Fixture	Verdict	Queries	Output or reason
three junctions, factors (9, 1), (9, 1), (2, 1)	SAT	45	(37859/40960, 37379/40960, 7373/8192), slack 1/1024
same graph, B private on the path $A \rightarrow B \rightarrow C$	SAT	31	$x_B = 6746098527/7381975040$, $\lambda = 1/4$ after two bisections
unit factors	UNSAT	1	Proposition 4.3(ii)
singleton boxes at (17)	SAT	15	the state (17) itself, slack 1/8192
directed cycle	UNSAT	1	Proposition 4.3(i)
path listed against its orientation	INVALID	0	orientation/discontinuity
box with lower bound above upper bound	INVALID	0	invalid junction box
four junctions with bound $k \leq 3$	OUTSIDE	0	class bound exceeded

Table 2: Eight-fixture regression of the executable demonstration. Every SAT output was rechecked exactly. The first output is a different rational witness from (17) for the same assembly; the second reconstructs the private species with certified last-edge margin 4137432795063009099/4219980937237211643904000 and a proved budget of 21 iterations.

6.8 An executable demonstration

The proofs above specify the one-block elimination routine as the decision backend. A small executable in the accompanying repository answers the same reduced formulas with the exact nonlinear real arithmetic solver of Z3 [15, 16] in place of that routine, performs the dyadic slack search, the bracket bisection and the private bisection exactly as in Steps 3–5, rechecks every output with exact rational arithmetic, and reports UNKNOWN rather than a verdict when a query times out. The worst-case bound of Theorem 6.1 is not asserted for this backend. Table 2 lists its regression on eight fixtures for the graph of Figure 3; the run used 93 queries and about 0.15 seconds in total (Windows, Python 3.11, Z3 4.15, a 3-second per-query limit), which is a demonstration measurement rather than a benchmark.

Two further small pilots test the lemmas rather than the theorem. The representation recurrence (13) was run up to $h = 8$: the degree and denominator bit length were exactly 2, 4, 16, 256 for $h = 1, 2, 4, 8$ with unit factors, and 52, 144, 504 denominator bits for $h = 4$ with input heights $B = 2, 8, 32$, matching Lemma 6.2. The bisection of Lemma 6.5 was run on eighteen paths of lengths 2, 3, 5 with target endpoints at relative positions 10^{-4} , $1/2$ and $1 - 10^{-4}$ of the band; the largest observed iteration count was 14 against a largest proved budget of 43. Narrowing the band of the three-junction instance by factors 10, 10^3 and 10^5 raised the query count from 15 to 22 to 28, with slacks 2^{-13} , 2^{-20} and 2^{-26} . None of this is evidence for the universal statements, which are proved above; it checks that the implemented recurrences and searches behave as the proofs predict.

7 A constructive linear sufficient class

The exact residual identities also yield a fast sufficient construction that does not need the junction–path structure at all. Write activities as deficits from the equilibrium point,

$$x_s = 1 - \varepsilon d_s, \quad d_s > 0, \quad \alpha_e = \frac{a_e + 2b_e}{a_e + b_e}, \quad \beta_e = \frac{a_e + 4b_e}{a_e + 2b_e}.$$

Note $1 < \alpha_e < \beta_e < 2$.

Proposition 7.1 (Finite-deficit certificate). *Suppose $\alpha_e d_u < d_v < \beta_e d_u$ on every selected edge $e: u \rightarrow v$, and put*

$$U_e = (a_e + 4b_e)d_u - (a_e + 2b_e)d_v > 0, \quad V_e = (a_e + b_e)d_v - (a_e + 2b_e)d_u > 0.$$

If

$$0 < \varepsilon < \min_e \frac{U_e}{2b_e d_u^2} \quad \text{and} \quad \varepsilon \max_s d_s \leq 1 - \ell, \quad (15)$$

then x lies in $[\ell, 1]^V$ and makes every selected core productive. A junction box $[l_s, u_s]$ additionally requires $(1 - u_s)/d_s \leq \varepsilon \leq (1 - l_s)/d_s$.

Proof. Substituting into (4) gives the exact expansions, with no discarded remainder,

$$r_{e,u} = \varepsilon U_e - 2b_e \varepsilon^2 d_u^2, \quad r_{e,v} = \varepsilon V_e + b_e \varepsilon^2 d_u^2. \quad (16)$$

The first is positive exactly when $\varepsilon < U_e/(2b_e d_u^2)$, the second whenever $V_e > 0$. The box conditions are direct. $\square$

For rational data the strict homogeneous system $\alpha_e d_u < d_v < \beta_e d_u$, $d_s > 0$ is decided by one rational linear program: normalize $\sum_s d_s = 1$, maximize t subject to $d_s \geq t$, $d_v - \alpha_e d_u \geq t$ and $\beta_e d_u - d_v \geq t$. The optimum is positive exactly when the strict system is feasible, and an optimal basic solution is rational of polynomial bit length [17, 18]. The bounds (15) and the box constraints are then finitely many rational intervals in ε , whose intersection is computed in polynomial time. Without junction boxes the intersection is nonempty whenever the linear program is positive; with junction boxes it may be empty for the returned deficits, and the procedure is then a sufficient test for those deficits rather than an optimization over all of them. A failed certificate does not prove incompatibility. Since $\alpha_e > 1$, a directed cycle never passes.

A nongraded unit-factor assembly. With $a_e = b_e = 1$ the ratio interval is $(3/2, 5/3)$. Two paths between the same pair of junctions with lengths $r < r'$ can both satisfy the strict ratio criterion only if $(3/2)^{r'} < (5/3)^r$; lengths 3 and 4 fail, and 4 and 5 are the first adjacent lengths that succeed, since $(3/2)^5 = 7.59 \dots < 7.71 \dots = (5/3)^4$. Take

$$d^{(4)} = (1, \frac{83}{50}, \frac{69}{25}, \frac{459}{100}, \frac{191}{25}),$$

$$d^{(5)} = (1, \frac{1501}{1000}, \frac{1127}{500}, \frac{3383}{1000}, \frac{1269}{250}, \frac{191}{25}),$$

sharing their first and last entries, and $\varepsilon = 10^{-4}$. Every adjacent ratio lies strictly in $(3/2, 5/3)$, all activities lie in $[9/10, 1]$, and exact evaluation of (16) over the nine cores gives minimum production $21/10^8 > 0$. The assembly has nine species (two junctions and seven private species), nine cores and two junctions (Figure 2). An integer rank increasing by one on every core would assign both 4 and 5 to the same pair of junctions, so no grading exists, and the graded certificate of [4] cannot produce this state. No minimality claim is made for this example; the six-and-seven-edge version was found first.

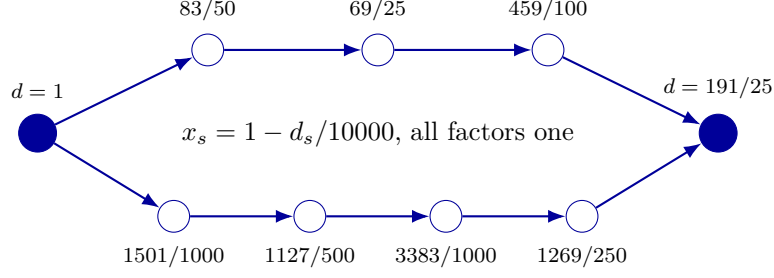


Figure 2: The unit-factor four/five assembly. Labels are the exact deficits; the two junctions are shared. The displayed rational state makes all nine cores productive in $[9/10, 1]$ with minimum production $21/10^8$, although no integer grading of the graph exists.

Edge	p_e	q_e	$2q_e - p_e$	$p_e - q_e$	ρ_e
AB	$\frac{3}{200}$	$\frac{247}{30000}$	$\frac{11}{7500}$	$\frac{203}{30000}$	$\frac{11}{236}$
BC	$\frac{7}{400}$	$\frac{3451}{360000}$	$\frac{301}{180000}$	$\frac{2849}{360000}$	$\frac{43}{943}$
AC	$\frac{13}{1800}$	$\frac{283}{45000}$	$\frac{241}{45000}$	$\frac{7}{7500}$	$\frac{21}{304}$

Table 3: Exact currents, residuals and per-edge relative-factor radii ρ_e of Proposition 8.1 at the state (17), food activity one. Every residual is strictly positive.

8 Operating certificates at a weighted example

Throughout this section use the three cores $A \rightarrow B$, $B \rightarrow C$ and $A \rightarrow C$ of Proposition 4.3(ii), now with factors and state

$$(a, b)_{AB} = (9, 1), \quad (a, b)_{BC} = (9, 1), \quad (a, b)_{AC} = (2, 1), \quad (17)$$

$$(x_A, x_B, x_C) = \left(\frac{99}{100}, \frac{593}{600}, \frac{3551}{3600} \right).$$

All activities lie in $[9/10, 1]$. Table 3 gives the exact currents; every residual is positive, with minimum $7/7500$ on the target side of the shortcut. The state also arises from Proposition 7.1 with $d = (1, 7/6, 49/36)$ and $\varepsilon = 1/100$, whose U_e, V_e are positive and satisfy (15). Figure 3 shows the composed responses of the two-edge path and of the shortcut: their bands overlap, whereas with unit factors they never do by (10). The rank equations $r_B = r_A + 1$, $r_C = r_B + 1$, $r_C = r_A + 1$ are inconsistent, so this compatible assembly is also nongraded.

Holding all three activities, $b_{AC} = 1$ and the two other cores fixed, the shortcut is productive exactly when $q_{AC} < a_{AC}(x_A - x_C) < 2q_{AC}$, i.e.

$$\frac{566}{325} < a_{AC} < \frac{1132}{325}.$$

This slice does not contradict the unit-factor incompatibility, which changes a_{AB} and a_{BC} as well.

8.1 Sharp relative-factor tolerance

Proposition 8.1 (Independent relative uncertainty). *Fix a productive state with nonempty edge set and let each factor vary independently in $[(1 - \rho)a_e, (1 + \rho)a_e]$ and $[(1 - \rho)b_e, (1 + \rho)b_e]$,*

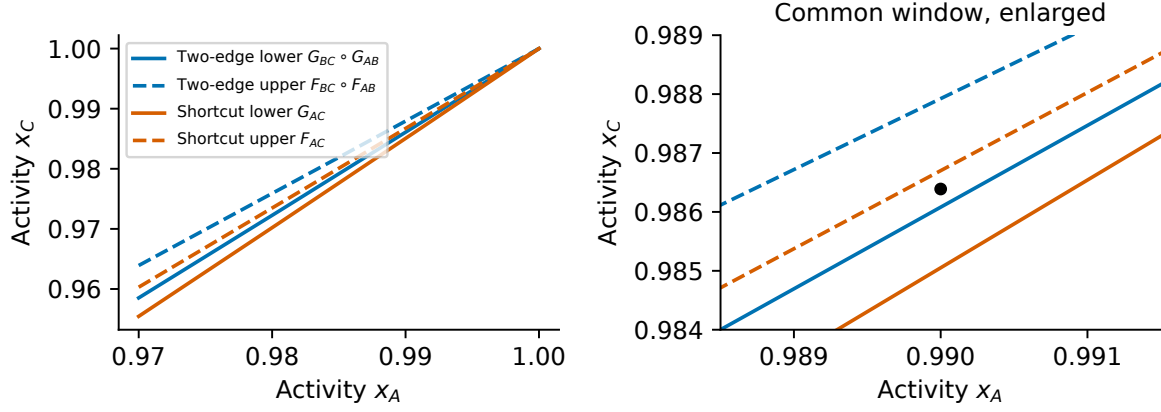


Figure 3: Composed responses for the weighted example. The band of the two-edge path $A \rightarrow B \rightarrow C$ (blue) and the band of the shortcut $A \rightarrow C$ (orange) overlap; the black point is the exact endpoint pair (x_A, x_C) of (17). Curves are sampled for display from the exact formulas; the certificate is rational.

$0 \leq \rho < 1$. Every core stays strictly productive throughout the whole factor box if and only if $\rho < \rho_*$, where

$$\rho_* = \min_{e \in E} \rho_e, \quad \rho_e = \min \left\{ \frac{2q_e - p_e}{2q_e + p_e}, \frac{p_e - q_e}{p_e + q_e} \right\}, \quad (18)$$

with p_e, q_e the nominal currents.

Proof. With factor multipliers $u, v \in [1 - \rho, 1 + \rho]$ the residuals become $2vq_e - up_e$ and $up_e - vq_e$, affine in (u, v) with $p_e, q_e > 0$. The first is minimized at $(u, v) = (1 + \rho, 1 - \rho)$ with value $2q_e - p_e - \rho(2q_e + p_e)$, the second at the opposite corner with value $p_e - q_e - \rho(p_e + q_e)$. Both minima are positive exactly when $\rho < \rho_e$, and at $\rho = \rho_e$ a corner has a zero residual, so the threshold is attained. $\square$

At (17) the three radii are $11/236$, $43/943$ and $21/304$ (Table 3), so $\rho_* = 43/943 \simeq 0.0456$, set by the source side of $B \rightarrow C$. At $\rho = 1/25$ every residual throughout the factor box is at least $77/375000$. For absolute perturbations $|a'_e - a_e|, |b'_e - b_e| \leq \rho$ one has $|p' - p|, |q' - q| \leq \rho$ on $[0, 1]^2$, hence residual changes at most 3ρ ; at $\rho = 10^{-4}$ every residual stays at least $7/7500 - 3 \cdot 10^{-4} = 19/30000$. Sharpness concerns the fixed state; a state reoptimized after perturbation could tolerate more.

8.2 Joint activity, factor and food rectangles

Let activities range independently in positive intervals $[L_s, U_s]$, factors in positive intervals, and the food activity in $[f_-, f_+]$ with $f_- > 0$. Then $p = a(x - y)$, $q = b(fy - x^2)$ and

$$r_u = -2bx^2 - ax + (a + 2bf)y, \quad r_v = bx^2 + ax - (a + bf)y. \quad (19)$$

Proposition 8.2 (Eight-corner certificate). *Every core is productive throughout the whole rectangle if and only if, for every edge $e: u \rightarrow v$, $r_u > 0$ at $(x, y, f) = (U_u, L_v, f_-)$ for each of the four factor corners, and $r_v > 0$ at $(x, y, f) = (L_u, U_v, f_+)$ for each of the four factor corners.*

Proof. For nonnegative factors, activities and food, r_u is nonincreasing in x and nondecreasing in y and f , and r_v has the opposite monotonicities, so each is minimized over the activity/food rectangle at the stated corner. At fixed (x, y, f) each residual is affine in (a, b) , so its minimum

over the factor rectangle is at a corner. Sufficiency follows. Conversely a nonpositive edge corner is a point of the common rectangle, obtained by choosing all other coordinates arbitrarily, at which that core is not productive; shared species do not obstruct this because one violating core suffices. $\square$

With food fixed at one, factors within $\pm 4\%$ of (17) and all activities within $\pm 10^{-6}$, the least of the 24 corner residuals is

$$\frac{847959991}{4687500000000} > 0, \quad (20)$$

attained on the source side of $B \rightarrow C$ at $a = 234/25$, $b = 24/25$, $x_B = 2965003/3000000$, $x_C = 8877491/9000000$. This certifies a *simultaneous* preparation-and-factor box; it is not a combination of separate maximal tolerances, and finding a largest such rectangle is a different problem.

8.3 Food windows, scaling and loss budgets

At fixed positive x, y and factors with $p = a(x - y) > 0$, solving $p/2 < b(fy - x^2) < p$ gives the exact food interval

$$\frac{x^2 + p/(2b)}{y} < f < \frac{x^2 + p/b}{y}. \quad (21)$$

Intersecting over the three cores of (17) gives

$$\frac{14814}{14825} < f < \frac{88859}{88775}; \quad (22)$$

the lower end makes the source residual of $A \rightarrow B$ vanish and the upper end the target residual of $A \rightarrow C$. Raising food at a *fixed* state overdrives the second channel and consumes its intermediate too quickly for core-wise productivity (Figure 4, right). This says nothing about feed being harmful when the state is allowed to respond. For general $f > 0$ the substitution $x_s = fz_s$ gives the exact scaling

$$p_e = f a_e(z_u - z_v), \quad q_e = f(b_e f)(z_v - z_u^2), \quad (23)$$

so a food activity f is equivalent to unit food with factors $(a_e, b_e f)$ and boxes divided by f ; Theorem 5.1 applies whenever the transformed boxes satisfy its hypotheses.

Let P_s be the sum of the incident production components at species s . At (17),

$$(P_A, P_B, P_C) = \left(\frac{307}{45000}, \frac{1519}{180000}, \frac{637}{72000} \right), \quad \left(\frac{P_A}{x_A}, \frac{P_B}{x_B}, \frac{P_C}{x_C} \right) = \left(\frac{307}{44550}, \frac{1519}{177900}, \frac{637}{71020} \right).$$

Each second channel consumes one food unit per internal unit produced, so total internal production and total food consumption both equal $\sum_e q_e = 2893/120000$. For the explicitly specified loss model $\dot{x}_s = P_s(x, f) - (\mathcal{D} + d_s)x_s$ with dilution $\mathcal{D}$ and species-specific degradation $d_s \geq 0$, every species increases at the instant the state (17) is prepared exactly when

$$\mathcal{D} < \min_s \{P_s/x_s - d_s\}, \quad (24)$$

which for $d \equiv 0$ is $\mathcal{D} < 307/44550 \simeq 0.00689$ in the normalized inverse-time unit. This is a statement about one instant of one specified ODE. It establishes neither an invariant region nor a positive steady state nor growth over any finite time.

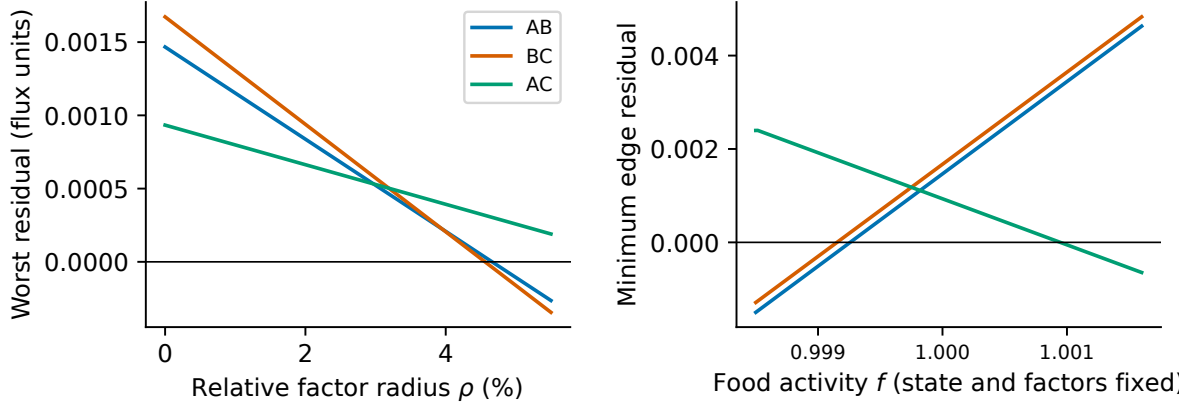


Figure 4: Two separate slices at the weighted state. Left: the exact affine lower envelopes of the residuals under independent relative factor uncertainty, food fixed at one; the first zero is at $\rho_* = 43/943$. Right: the edgewise minimum residual as the food activity varies with state and factors fixed; the window is (22). Neither panel is the joint certificate (20).

9 Formal verification

The statements marked as compiled in Table 4 are Lean 4 theorems [7] in the namespace `ThermoCoreCompatibility.MultiInterface`, checked against Mathlib [8] with toolchain `leanprover/lean4:v4.30.0` and a frozen dependency manifest. Verification promotes every warning to an error, rejects `sorry`, probes the requested declarations, records the axioms each depends on, and hashes the sources. Every verified declaration depends only on `propext`, `Classical.choice` and `Quot.sound`. Two receipts are kept: the root receipt for the source equivalence (module `Root.lean`, source SHA-256 99270f83..., dependency manifest a8df0b60..., 67 seconds), and the extended receipt for the certificate lemmas and examples (module `PostproofRoot.lean`, source SHA-256 c0f3b02a..., 102 seconds), whose dependency closure includes the unchanged root.

Architecture. The development comprises about 1,000 lines in 17 files and builds on the reversible-network semantics of [3] and the pair-assembly source adapter of [4].

- `WeightedPair.lean`: the structure `Factors` of positive (a, b) , the responses, the identity `Factors.productive_iff` (Lemma 3.1), strict monotonicity, the gap formula, $x^2 < G < F < x$, and continuity.
- `PathResponse.lean`, `PathReconstruction.lean`, `PathCoordinates.lean`, `PathEdges.lean`: the inductive `Path` built by appending edges in physical order, the composed responses, the realization predicate, the interpolation orbit, the intermediate-value construction `Path.sufficient`, and the boxed equivalence `Path.boxed_iff` (Theorem 4.1); explicit vertex and edge coordinates for a path.
- `JunctionAssembly.lean`: the structure recording paths, vertex placement, the junction predicate, the endpoint characterization and private uniqueness; the merge lemma `JunctionAssembly.merge` and the abstract equivalence `compatible_iff_feasible`.
- `WeightedSource.lean`, `JunctionSource.lean`: the weighted reversible network on the literal channels (2) with per-channel barrier factors, the rebasing of each core motif with preserved PAC status, the identification of source-level productivity with `Factors.Productive` (`current_productive_iff`), the simple-graph structure `SourceAssembly`, and the theorem `sourceCompatible_iff_junctionFeasible` (Theorem 5.1).

Statement	Declaration or proof location	Evidence
One-edge interval, Lemma 3.1	<code>Factors.productive_iff</code> , <code>Factors.gap</code> , <code>Factors.lower_strictMono</code>	Lean
Path reconstruction, Theorem 4.1	<code>Path.realizes_iff</code> , <code>Path.boxed_iff</code>	Lean
Merge and assembly, Theorem 5.1	<code>JunctionAssembly.merge</code> , <code>sourceCompatible_iff_junctionFeasible</code>	Lean
Obstructions, Proposition 4.3	<code>Factors.decreases</code> ; <code>Examples.unit_source_incompatible</code>	Lean
Rational checker	<code>rationalFamilyCheck_sound</code>	Lean
Representation, Lemma 6.2	Section 6; recurrence pilot	Conventional
Slack, precision, bisection, Lemmas 6.3–6.5	Section 6; backend [5, Thm. 2.27]	Conventional
Bit bound, Theorem 6.1	Section 6	Conventional
Finite deficit, Proposition 7.1	<code>deficit_identities</code> , <code>deficit_productive</code> , <code>deficit_box</code>	Lean
LP complexity	Section 7, [17]	Conventional
Four/five assembly, nongraded	<code>PostproofExamples.unit_source</code> , <code>PostproofExamples.nongraded</code>	Lean
Relative radius, Proposition 8.1	<code>relative_box_iff</code> , <code>relative_radius_iff</code>	Lean
Eight corners, Proposition 8.2	<code>activity_corner</code> , <code>factor_corners</code> ; composition in the proof	Lean lemmas
Food, scaling, loss	<code>food_window</code> , <code>food_scaling</code> , <code>instantaneous_loss</code>	Lean
Printed operating numbers	<code>PostproofExamples.weighted_relative</code> , <code>joint_corner_values</code> ; exact replay script	Lean and exact replay

Table 4: Theorem/evidence concordance. Lean names are relative to the namespace of the development.

- `RationalWitness.lean`, `Examples.lean`, `PostproofExamples.lean`: the decidable rational checker `rationalFamilyCheck` and its soundness theorem at source level, the weighted example (17) (`Examples.weighted_source`), its unit-factor incompatibility, the four/five assembly (`PostproofExamples.unit_source`), both nongradedness proofs, and exact evaluations of the radii, the 4% residual bound and all 24 joint corners.
- The four certificate modules `RelativeRobustness.lean`, `OperatingBox.lean`, `FoodWindow.lean` and `LinearCertificate.lean`: Propositions 8.1 and 8.2 (activity/food monotonicity and affine factor corners), the food window (21), the scaling (23), the instantaneous-loss equivalence (24), and the deficit identities (16) with Proposition 7.1.

Boundary. Three things are conventional and are not claimed to be verified: the instantiation of the external quantifier-elimination bound, the bit-size accounting of Section 6, and the polynomial complexity of rational linear programming. The executable demonstration of Section 6.8 is not a verified program, and its Python validator and data encoder are trusted only through the checking argument of Step 1; a supplied SAT state, however, is validated by the compiled soundness theorem of the rational checker once its data are transcribed. Every rational number printed in Sections 7 and 8 is replayed by a standard-library script distributed with the source.

10 Discussion

What the reduction shows. Compatibility of a junction–path assembly is a property of its junction activities alone, and the property is explicit: two composed quadratic responses per path. Nothing about the interior is lost, because the interpolation orbit realizes every value strictly between the composed responses. This is stronger than a relaxation: the linear relaxations of [3, 4] discard the equation $y_{2X_u} = x_u^2$, whereas the path reduction keeps it and discards only variables that are provably recoverable.

The role of the factors. The unit-factor shortcut is incompatible for every positive state, while factors $(9, 1), (9, 1), (2, 1)$ admit a common state with all residuals positive and a 4.6% independent factor tolerance. Since compatibility depends only on the ratios a_e/b_e , the picture is that of a response band per path whose width scales with $a_e b_e / ((a_e + b_e)(a_e + 2b_e))$; a shortcut with a slower first channel relative to its doubling channel has a lower band that can dip below the upper band of a longer path. Whether a given topology admits *some* factor assignment making it compatible is a question on the closure of the union of the bands and is not addressed here.

Limits. The decomposition is input; minimizing k over decompositions, or recognizing whether an assembly with a small k exists, is not treated. Private species share one box, and Remark 4.2 shows that independent private boxes break the two-function reduction. Cores are the two-reaction gain-two motif; higher gains and cores that share physical reactions are outside the class. The complexity bound is singly exponential in k through the elimination routine and depends exponentially on h through the expanded degree; neither dependence is shown to be necessary. The operating certificates quantify a fixed state; they do not assert that the state is reached, maintained or selected by any dynamics.

Open questions.

- (1) Is there a decision procedure for junction–path assemblies whose cost is polynomial in h , for instance by working with the recursive representation of G_P, F_P instead of the expanded polynomial?
- (2) For which oriented simple graphs does some positive factor assignment make the whole edge family compatible? Directed cycles never do; the shortcut does; a characterization is open.
- (3) Can arbitrary private boxes be handled by introducing at most one additional variable per boxed private species, and does the resulting exponent remain $O(k' + 1)$ in the enlarged variable count k' ?
- (4) Does a common productive state of a junction–path assembly persist under the mass-action dynamics with the food chemostatted and a dilution below (24), and if so on what time scale?

Reproducibility. The source of this paper, the bibliography, the figure generator, the exact replay script and the data it reads are distributed together; the Lean development, the verification receipts, the executable demonstration and the conventional proof notes are in the accompanying repository.

References

- [1] Alex Blokhuis, David Lacoste, and Philippe Nghe. Universal motifs and the diversity of autocatalytic systems. *Proceedings of the National Academy of Sciences*, 117(41): 25230–25236, 2020. doi:10.1073/pnas.2013527117.
- [2] Thomas Kosc, Denis Kuperberg, Etienne Rajon, and Sylvain Charlat. Thermodynamic consistency of autocatalytic cycles. *Proceedings of the National Academy of Sciences*, 122(18):e2421274122, 2025. doi:10.1073/pnas.2421274122.
- [3] Anonymous. Thermodynamic compatibility of interacting autocatalytic cores: an exact phase diagram and a layered obstruction theory, 2026. Preprint, September 2026. Formal development in the same repository as the present paper.
- [4] Anonymous. Collective thermodynamic incompatibility of autocatalytic cores: minimal conflicts of unbounded order, bounded fans, and graded assemblies, 2026. Preprint, September 2026. Formal development in the same repository as the present paper.
- [5] Saugata Basu. Algorithms in real algebraic geometry: a survey. arXiv preprint arXiv:1409.1534 [math.AG], 2014. arXiv:1409.1534. Theorem 2.27 is the statement used here.
- [6] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer, Berlin, second edition, 2006. doi:10.1007/3-540-33099-2.
- [7] Leonardo de Moura and Sebastian Ullrich. The Lean 4 theorem prover and programming language. In *Automated Deduction – CADE 28*, volume 12699 of *Lecture Notes in Computer Science*, pages 625–635. Springer, 2021. doi:10.1007/978-3-030-79876-5_37.
- [8] The mathlib Community. The Lean mathematical library. In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020)*, pages 367–381. ACM, 2020. doi:10.1145/3372885.3373824.
- [9] Daniel A. Beard, Eric Babson, Edward Curtis, and Hong Qian. Thermodynamic constraints for biochemical networks. *Journal of Theoretical Biology*, 228(3):327–333, 2004. doi:10.1016/j.jtbi.2004.01.008.
- [10] Dima Yu. Grigor’ev and Nicolai N. Vorobjov. Solving systems of polynomial inequalities in subexponential time. *Journal of Symbolic Computation*, 5(1–2):37–64, 1988. doi:10.1016/S0747-7171(88)80005-1.
- [11] James Renegar. On the computational complexity and geometry of the first-order theory of the reals. Parts I–III. *Journal of Symbolic Computation*, 13(3):255–352, 1992. doi:10.1016/S0747-7171(10)80003-3.
- [12] George E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages*, volume 33 of *Lecture Notes in Computer Science*, pages 134–183. Springer, 1975. doi:10.1007/3-540-07407-4_17.
- [13] Alfred Tarski. *A Decision Method for Elementary Algebra and Geometry*. University of California Press, Berkeley, second edition, 1951.
- [14] Maurice Mignotte. *Mathematics for Computer Algebra*. Springer, New York, 1992. doi:10.1007/978-1-4613-9171-5.

- [15] Leonardo de Moura and Nikolaj Bjørner. Z3: an efficient SMT solver. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2008)*, volume 4963 of *Lecture Notes in Computer Science*, pages 337–340. Springer, 2008. doi:[10.1007/978-3-540-78800-3_24](https://doi.org/10.1007/978-3-540-78800-3_24).
- [16] Dejan Jovanović and Leonardo de Moura. Solving non-linear arithmetic. In *Automated Reasoning (IJCAR 2012)*, volume 7364 of *Lecture Notes in Computer Science*, pages 339–354. Springer, 2012. doi:[10.1007/978-3-642-31365-3_27](https://doi.org/10.1007/978-3-642-31365-3_27).
- [17] Leonid G. Khachiyan. A polynomial algorithm in linear programming. *Doklady Akademii Nauk SSSR*, 244(5):1093–1096, 1979. English translation: Soviet Mathematics Doklady 20 (1979), 191–194.
- [18] Alexander Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Chichester, 1986.